

INSTITUTO SUPERIOR de ENGENHARIA de LISBOA

Licenciatura em Engenharia Informática e Multimédia

1º Semestre Letivo 2022/2023

Fundamentos de Sistemas Operativos

1º Trabalho Prático

Objetivos:

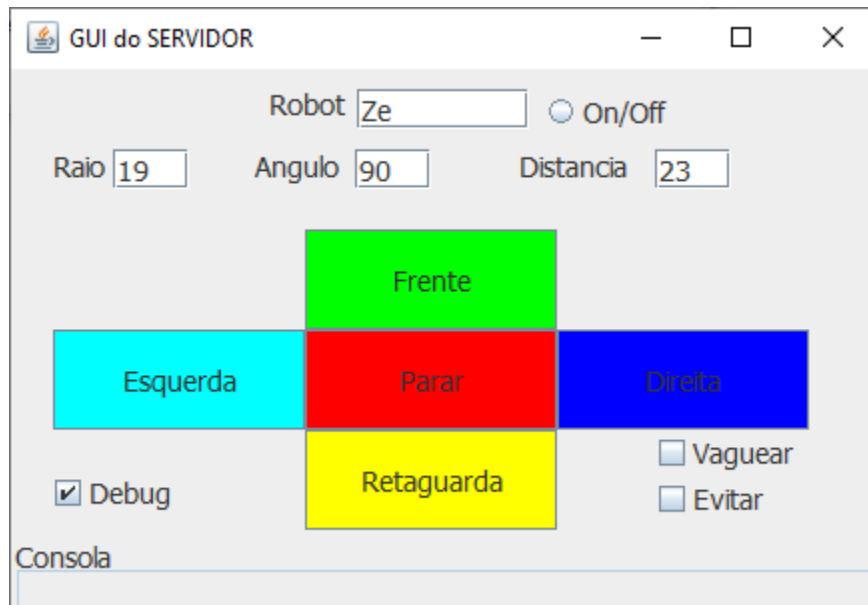
- Desenvolvimento de interfaces gráficas em JAVA Swing utilizando o editor gráfico WindowBuilder para o Eclipse.
- Gestão de processos JAVA.
- Comunicação entre processos JAVA com memória partilhada.
- Modelo Cliente-Servidor.

Pretende-se o desenvolvimento de três processos JAVA, um processo implementa o comportamento VAGUEAR, o segundo processo implementa o comportamento EVITAR e o terceiro processo é um SERVIDOR que conhece a API do robot. Os processos VAGUEAR e EVITAR são clientes do processo SERVIDOR.

A comunicação entre os processos utiliza canais de comunicação implementados com base em memória partilhada suportada pela classe `MappedByteBuffer`. O canal de comunicação deve conter $N+1$ mensagens até ficar cheio, em que N é o “vosso” número de grupo.

O **protocolo de comunicação**, será especificado por cada grupo e é definido por um conjunto de mensagens, suportando a implementação do modelo **cliente-servidor**.

Adicione à interface gráfica do SERVIDOR, duas *checkboxes* “Vaguear”, “Evitar”, onde cada *checkbox* é responsável por ativar ou desativar o respetivo comportamento.



O comportamento VAGUEAR é um processo JAVA que faz o robot vaguear pelo espaço. A definição de vaguear no espaço consiste no robot andar em frente, curvar à direita, curvar à esquerda e parar de forma aleatória. Este comportamento fica ativo ou desativo função da GUI. Este comportamento deve ter uma GUI com uma consola para *debugging*.

O comportamento EVITAR consiste num processo JAVA que quando ativo, pede o valor do sensor de toque ao SERVIDOR de 250ms em 250ms e se o valor for 1 então o robot deve parar imediatamente e depois anda 15cm para trás, de seguida faz um curvar à esquerda com raio de 0cm e ângulo de 90 graus e depois pára o robot. Se o sensor de toque continuar ativo deve repetir o procedimento anterior até o sensor ficar desativo. Este comportamento deve ter uma GUI com uma consola para debugging.

O processo SERVIDOR é o único processo que conhece a API do robot, daí ser o processo designado por **servidor**. Os processos VAGUEAR e EVITAR são processos designados como **cliente**. O modelo **cliente-servidor** define que os processos **cliente** têm de comunicar com o processo **servidor** sempre que querem interagir com o robot.

A implementação do comportamento SERVIDOR consiste num processo JAVA que utiliza objetos da classe “CanalDeComunicação” para comunicar com os processos cliente. Cada objeto da classe “CanalDeComunicação” serve para comunicar unidireccionalmente entre o processo servidor e um processo cliente.

Sugestões para o desenvolvimento do trabalho:

1. Identifique os comandos que cada um dos processos necessita de enviar e receber. Transforme cada comando numa mensagem. O conjunto de mensagens resultante define o protocolo de comunicação. Este protocolo de mensagens deve estar especificado numa interface “Mensagem”.
2. Crie uma classe "CanalDeComunicação" que disponibiliza os métodos necessários para a troca de mensagens entre os diferentes processos.
3. Todos os processos devem ter objetos da classe “CanalDeComunicação” para a concretização da comunicação entre processos ponto a ponto.

Avaliação

A duração do trabalho prático é de **6** aulas práticas. O trabalho é realizado em grupo e está sujeito a avaliação.

A avaliação do trabalho prático tem em conta os seguintes aspetos:

1. Estrutura e organização do relatório;
2. Código implementado;
3. Descrição dos protocolos de comunicação;
4. Diagramas de atividade ou de estado de cada processo.

O relatório do trabalho prático e o código implementado devem ser entregues uma semana após a realização da última aula prática. A discussão será realizada na semana seguinte à

semana de entrega do relatório.

Objetivos a cumprir nas aulas práticas

1ª aula prática – Desenho da GUI (*Graphical User Interface*) do processo SERVIDOR.

2ª aula prática - Especificação do protocolo de comunicação. Implementação e teste da interface “Mensagem” e da classe “CanaldeComunicação”.

3ª aula prática – Desenho da arquitetura da aplicação onde especifica todos os processos e respetivos canais de comunicação. Desenho do diagrama de atividades do comportamento SERVIDOR tendo em consideração a arquitetura proposta. Implementação e teste.

4ª aula prática – Desenho do diagrama de atividades do comportamento VAGUEAR. Implementação e teste deste comportamento conjuntamente com o processo SERVIDOR.

5ª aula prática – Desenho do diagrama de atividades do comportamento EVITAR. Implementação e teste deste comportamento conjuntamente com o processo SERVIDOR.

6ª aula prática – Teste da aplicação multiprocesso completa. Demonstração ao Professor e respetiva validação.

Os Docentes,
Jorge Pais, Carlos Gonçalves e Carlos Carvalho